

**aThis Page Is Inserted by IFW Operations
and is not a part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

Requested Patent JP11085540A

Title:

DEVICE AND METHOD FOR DYNAMIC REGULATION OF THE RESOURCE
ALLOCATION IN A COMPUTER SYSTEM ;

Abstracted Patent EP0893761 ;

Publication Date: 1999-01-27 ;

Inventor(s):

SITBON GERARD (FR); URBAIN FRANCOIS (FR); DURAND DANIEL LUCIEN
(FR) ;

Applicant(s): BULL SA (FR) ;

Application Number: EP19980401854 19980721 ;

Priority Number(s): FR19970009369 19970723 ;

IPC Classification: G06F9/46 ;

Equivalents: FR2766592

ABSTRACT:

The method involves sorting the work into dimensions. A weight relative to each of the dimensions is allocated by the user. When the system is heavily loaded, the work execution priorities of each dimension are modulated according to their relative weight. When the system uses the hardware resources above a high threshold, the method will dynamically modify the priority associated with the work in progress, according to the dimensions to which it belongs. When the load of hardware use goes back below this threshold the original work priorities are re-established.

(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平11-85540

(43)公開日 平成11年(1999)3月30日

(51)Int.Cl.⁶

G 0 6 F 9/46

識別記号

3 4 0

F I

G 0 6 F 9/46

3 4 0 D

審査請求 有 請求項の数7 OL (全14頁)

(21)出願番号 特願平10-207006

(22)出願日 平成10年(1998)7月22日

(31)優先権主張番号 97 09369

(32)優先日 1997年7月23日

(33)優先権主張国 フランス (FR)

(71)出願人 390035633

ブル・エス・アー

フランス国、エフー78434・ルーブシエン
ヌ、ルート・ドウ・ベルサイユ、68

(72)発明者 ダニエル・リュシヤン・デュラン

フランス国、78180・モンティニー・ル・
ブルトヌー、リュ・マリア・カラース、38

(72)発明者 ジエラール・シトボン

フランス国、94400・ビトリー、リュ・ガ
ニエ、12

(72)発明者 フランソワ・ユルバン

フランス国、75002・パリ、リュ・ドウ・
パレストロ、33

(74)代理人 弁理士 川口 義雄 (外2名)

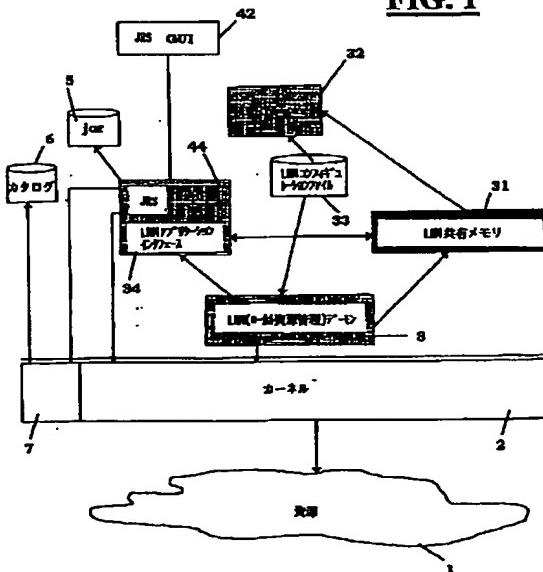
(54)【発明の名称】 情報処理システムにおける資源割り当ての動的制御のための装置および方法

(57)【要約】

【課題】 「ユニックス」型のオープン情報処理システムにおける資源割り当ての動的制御方法を提供する。

【解決手段】 この方法は、ジョブをディメンションで分類し、各ディメンションに関する相対的重み付けをユーザーが割り当て、システムの負荷が非常に大きい場合、ディメンションに関する重み付けに応じて各ディメンションのジョブの実行優先順位を調整することからなることを特徴とする。

FIG. 1



【特許請求の範囲】

【請求項1】 「 Unixx 」タイプのオープン情報処理システムにおける資源割り当ての動的制御方法であつて、

ジョブをディメンションでソーティングし、各ディメンションに関する相対的重み付けをユーザーに割り当てさせ、

システムの負荷が非常に大きい場合、ディメンションに関する相対的重み付けに応じて各ディメンションのジョブの実行優先順位を調整する、ことからなることを特徴とする動的制御方法。

【請求項2】 システムが、重要な閾値を越えるハードウェア資源を用いている場合、この方法は、ジョブが属するディメンションに応じて進行中のジョブに関する優先順位を動的に変えることを特徴とする、請求項1に記載の「 Unixx 」タイプのオープン情報処理システムにおける資源割り当ての動的制御方法。

【請求項3】 システムのハードウェア占有率が一定の閾値以下に下がる場合、この方法はジョブの初期の優先順位を再設定することを特徴とする、請求項1または2に記載の「 Unixx 」タイプのオープン情報処理システムにおける資源割り当ての動的制御法。

【請求項4】 「 Unixx 」タイプのオープン情報処理システムにおける資源割り当ての動的制御装置であつてジョブをディメンションでソーティングする手段と、

各ディメンションに関する相対的重み付けを割り当てる手段と、

システムの負荷が非常に大きい場合、ディメンションに関する相対的重み付けに応じて各ディメンションのジョブの実行優先順位を変える手段とを含むことを特徴とする動的制御装置。

【請求項5】 一定の時間間隔でシステムの状態を調べる手段と、ユーザーのジョブに様々なディメンションを関連付ける手段と、これらのディメンションの各々に大きい重み付けまたは小さい重み付けを与える手段とを含むことを特徴とする、請求項4に記載の「 Unixx 」タイプのオープン情報処理システムにおける資源割り当ての動的制御装置。

【請求項6】 システムは、ジョブが属するディメンションに応じて、進行中のジョブに関する優先順位を動的に変える手段を含むことを特徴とする、請求項4または5に記載の「 Unixx 」タイプのオープン情報処理システムにおける資源割り当ての動的制御装置。

【請求項7】 システムの負荷が一定の閾値以下に下がることを確認してジョブの初期の優先順位を再設定する手段を含むことを特徴とする、請求項4または6のいずれか一項に記載の「 Unixx 」タイプのオープン情報処理システムにおける資源割り当ての動的制御装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、情報処理システムにおける資源割り当ての動的制御のための装置ならびに方法に関する。

【0002】

【従来の技術】この発明は、工業環境における情報処理オペレーション分野、特に「 Unixx 」タイプソフトウェア上で動作する「オープン」型の情報処理システムに適用される。このタイプのソフトウェア上で動作するサーバは、いわゆる「オープン」型サーバであり、今日、特に重視されているものである。

【0003】これらのサーバの不都合は、目的が工業的な目的と同じではない大学のアプリケーションのために開発されたことにある。このように、「 Unixx 」タイプのOSでは、ジョブあるいはより正確には、ジョブを含む各プロセスは、システムまたは、これらのジョブを入力するユーザーによって初期の優先順位を割り当られる。システムが一つのジョブにより高い優先順位を与えるようにするために、ユーザーは、初期の優先順位をジョブに明白に割り当てるか、管理コマンドによってその優先順位を手動で変えなければならない。従って、ジョブに割り当てる優先順位レベルの間にはいかなる調整も相関関係もない。

【0004】同様に、ジョブの作業負荷に関し、システムの状態と、ジョブの優先順位の管理との間には自動的な相関関係がない。このように調整がない場合、全てのプロセスは同じ権利を有し、数の法則が優先する。M1, M2, M3 が重要ではない三つのプロセスグループであつて、システムプロセス S1, S2, S3 および重要な「ロット別」プロセス（バッチ）B1, B2, B3 よりも数が多くアクティブである場合、このプロセスグループは、資源の最大部分を占めることになる。従来の「 Unixx 」システムにおいて、プロセスは、非常に負荷の大きいシステムでも無制限に始動することができ、実行プロセスが現在使用している資源の数にはいかなる注意も払わず、新しいプロセスが必要とする資源の数にも全く留意しない。これによって次のような好ましくない挙動が起きる。すなわち重要なプロセスであろうと重要でないプロセスであろうと一連のプロセスが実行され、ユーザーが所望の時間に終了しなければならない重要なプロセスが使える資源は、そうした所望の時間に終了するには少なすぎる。同時に、重要なプロセスが用いる資源は多すぎる。

【0005】

【発明が解決しようとする課題】このような理由から、本発明の第一の目的は、これらの欠点を解消することができ、ジョブがシリアルまたはパラレルの一連のプロセスであるとき、プロセスおよびジョブの動的制御方法を提案することにある。

【0006】

【課題を解決するための手段】この目的は、「ユニックス」タイプのオープン情報処理システムにおけるプロセスの動的制御方法によって達せられ、この方法は、ジョブをディメンションでソーティングし、各ディメンションに関する相対的重み付けをユーザーに割り当てさせ、システムの負荷が非常に大きい場合、ディメンションに関する相対的重み付けに応じて各ディメンションのジョブの実行優先順位を調整することからなる。

【0007】他の特徴によれば、システムが、重要な閾値を越えるハードウェア資源を用いている場合、この方法は、ジョブが属するディメンションに応じて進行中のジョブに関する優先順位を動的に変える。

【0008】別の特徴によれば、システムのハードウェア占有率が一定の閾値以下に下がる場合、この方法はジョブの初期の優先順位を再設定する。

【0009】本発明の他の目的は、ジョブの動的制御装置を提案することにある。

【0010】この目的によれば、「ユニックス」オペレーションシステムにより実行されるジョブの動的制御装置は、ジョブをディメンションでソーティングする手段と、各ディメンションに関する相対的重み付けを割り当てる手段と、システムの負荷が非常に大きい場合、ディメンションに関する相対的重み付けに応じて各ディメンションのジョブの実行優先順位を変える手段とを含むことを特徴とする。

【0011】さらに別の特徴によれば、一定の時間間隔でシステムの状態を調べる手段と、ユーザーのジョブに様々なディメンションを関連付ける手段と、これらのディメンションの各々に大きい又は小さい重み付けを与える手段とを含む。

【0012】また別の特徴によれば、システムは、ジョブが属するディメンションに応じて、進行中のジョブに関する優先順位を動的に変える手段を含む。

【0013】もう一つの特徴によれば、装置は、システムのロードが一定の閾値以下に下がることを確認してジョブの初期の優先順位を再設定する手段を含む。

【0014】本発明の他の特徴ならびに長所は、添付図に関してなされた以下の説明を読めば、いっそう明らかになろう。

【0015】

【発明の実施の形態】以下、「プロセス」または「タスク」という表現は、所定の瞬間ににおけるプログラムのあらゆる実行（従って、特にシステムにおけるその環境）を示し、プログラムはそれ自体が実行可能な通常のファイルとしてディスクに格納された不活性物を構成する。「ユニックス」システムでは、次の二つのタイプのプロセスの存在が知られている。

【0016】どの端末にも結合せず、システムのスタートまたはシステムの管理者によって決定される日付で作成されるシステムプロセス。たとえば、プリンタの適切

な使用を保証するプロセス「スプール」または、所定の日にタスクをスタートさせられるプロセス「クロン」のような、一定数のいわゆる「デーモン」プロセスである「スワッパー」が挙げられる。

【0017】特定のユーザーにより所定の日付で所定の端末からスタートされるプロセス。特に、所定の識別のもとで端末に収容されることにより、各ユーザーに対して事前に決定された所定のファイルの実行に対応して、プロセスをスタートさせる。この種のプロセスは、殆ど常に制御言語のインターフリタ（ブルヌシェルまたはCシェル）の実行に対応する。参考として述べるが、遠距離にある機器でユーザーが標準サービスを利用する場合、遠距離にある機器において作動するため「デーモン」と称される特定プロセスの存在を必要とするコマンドを介する。

【0018】図1に示された「ユニックス」システムの構造は、情報処理システム1のハードウェア資源からなり、この情報システムは、メモリおよび低レベルの入出力および様々なタスク（ジョブ）の連続を管理する「ユニックス」カーネル（KERNEL）2と通信する。このカーネルの周囲で、「シェル」型制御言語の一つまたは複数の「インターフリター」、電子メールシステム、および様々なユーティリティープログラムが用いられ、ユーティリティープログラムには勿論、C言語のコンパイラを含む。

【0019】「ユニックス」システムにおけるこれらの従来の要素の周囲には、オープン環境にマルチタスクを適用する必要性から、二個のプログラムモジュールが展開される。その一方はユーザーによってスタートされたタスクの実行のレポートを提供する（JRS）であり、他方は、適切な経過あるいはトラブル発生をディスプレイし、従って原因を検出可能な（LRM）である。最初のモジュールは、ジョブリポートサービス「JRS」（Job reporting Service）と称され、後のモジュールは、ローカル資源管理モジュール「LRM」（Local Ressource Management）と称される。

【0020】さらに本発明の必要性から、カーネル2およびアプリケーションプログラムのインターフェース34と通信するローカル資源管理のデーモン3が付加される。デーモン3は、ハードディスク内に記憶されるLRMコンフィギュレーションファイル33および関連するグラフィカルユーザーインターフェース（LRM GUI）32とも通信する。アプリケーションプログラムのアプリケーションインターフェース34は、デーモン3および共有メモリ31と通信する。共有メモリ31はまた、グラフィカルユーザーインターフェース32およびデーモン3とも通信する。事前割り当ての必要性から、システムは、タスクの実行リポートを実施するアプリケーションプログラム「JRS」のインターフェース44

を組み込んでいる。このアプリケーション44は、ハードディスクの特別なファイル5にタスクリポートを記憶するカタログ「jor」と接続する。アプリケーションプログラムのインターフェース44は、プログラムが作動されていてタスクリポートファイルが記憶されているという条件で、ローカル資源管理デーモン3と通信し、資源の事前割り当ておよび動的制御を可能にする。インターフェース44は、このインターフェースによりユーザーが利用可能なコマンドの実行を可能にするために必要な数行のコードを含む。インターフェース44により利用されるコマンドは次の通りである。

【0021】— 資源の管理者にコマンドを送り、リポートファイルの実行を発生可能な「JRS」コマンド。このコマンドは、仮想メモリのページ数、一時的なファイルスペース量、ユーザーによって決定されるディメンション、予期されるCPUの消費時間、最大経過時間、最大CPU時間、一時的なファイルの最大スペース、仮想メモリページの最大数を決定可能な複数のオプションを含む。このコマンドは、上記の各種パラメータとともに後述する。

【0022】「JRS jobstart」コマンド(int jobid; int qflag; struct JRS_resources *JRS_resources;)、ここでjobidパラメータはタスクの識別子であり、JRS_resourcesはタスクに対して必要とされる資源を示す構造体である。タスクが資源の事前割り当てを必要としない場合、これは0となる。「qflag」は、それが0でないとき、タスクが資源の使用可能性を持つことを意味する。このコマンドは、タスクがスタートできるかどうか検証し、スタートできない場合には、資源が利用可能になるまで待つが、これは「qflag」が0のときである。「qflag」が0でない場合は、すぐに迅速な復帰を引き起こす。タスクがスタート可能である場合は値0が返され、そうでなければ「qflag」が設定され、値-1が返される。さらに、特定のタスクの主要な事象がカタログ「jor」のグローバルファイル「logc」に発生した場合、この事象は記憶される。タスクの各エントリに対してパラメータを明白に決定しないように、「JRS」アプリケーション44は、スクリプト内で資源宣言をする可能性を提供する。このようにアプリケーションの開発者がそのタスクに対する資源の必要性を適切に決定した場合、このタスクをスクリプトに挿入することができる。タスクは、他のいかなるパラメータも持たない「JRS」コマンドで入力される。このJRSコマンドの構文は以下の通りである。

【0023】

```
# option-p 「仮想メモリページ数」
# option-t 「一時的なファイルスペース量」
# option-d 「ユーザーによって決定される機
```

能」

```
# option-c 「予期されるCPU時間」
# option-e 「最大経過時間」
# option-C 「CPUによる消費時間の限界」
# option-P 「仮想メモリページの最大数」
# option-T 「一時的なファイルの最大スペース」
```

衝突がある場合には、コマンドのエントリで特定された値は、次のように用いられる。スクリプトの内部に次のような一行があり、

```
# option-C 200
```

スクリプトが次のコマンドラインと共に入力される場合、

```
JRS-C 300 script,
```

CPUの限界に対して用いられる値は300である。

【0024】このようにローカル資源管理モジュールを備えたシステムにおいて、重要でないプロセスは、重要なプロセスの実行の堆積ができるだけ変えないようにしなければならないだろう。各種のプロセスはその固有の重要性を持っている。こうした重要性に応じて、またシステムの負荷に応じて多かれ少なかれプロセスに資源を割り当たなければならないだろう。このような理由から「LRM」モジュールは、プロセスをディメンションでソートすることができる。一つのディメンションは、現在実行される一群のプロセスであり、ローカル資源管理の観点から同じ重要性を有することになる。デフォルトで五個のディメンションが設けられている。つまり第一のディメンション「システム(SYSTEM)」、第二のディメンション「ロット(LOT)」(バッチ)、第三のディメンション「ダイバース(DIVERS)」(MISC)、第四のデータベースディメンション(DB)、および第五の転送プロトコルディメンション(TP)である。通常のユーザーによって直接バックグラウンドでプロセスがスタートされる場合、プロセスはディメンション「ダイバース」に属する。いかなる明白なディメンションの宣言もなくタスク管理アプリケーションに入力されたスクリプトを介してプロセスがスタートされると、プロセスは、その場合、第二のディメンション「ロット」(バッチ)に属する。プロセス型のいかなるディメンションにも属さない場合、プロセスは、「システム」ディメンションに属する。ディメンション「DB」および「TP」は供給されても、いかなるプロセスの定義もなく、ユーザーによって定義されることになる。「システム」ディメンションを除いて、各ディメンションは、相対的な重み付けを有することが可能であり、この重み付けはファイルに記憶される。相対的な重み付けは、ディメンションを含むプロセスの優先順位を管理するために用いる。一つのディメンションに属するあらゆるプロセスの優先順位は、同じように変化する。「システム」ディメンションは、相対的な重み付けを持たず、

このディメンションに属するプロセスの優先順位は、ローカル資源管理（LRM）によって変えられない。ディメンションによって制御されるプロセスは、常に、たとえシステムの負荷が非常に大きくても、それらの相対的な重み付けに応じて与えられる資源である。ローカル資源管理を開始する場合、あるいは、相対的な重み付けを特定せずに一つのディメンションが形成される場合、このディメンションは、デフォルトで-1の重み付けを有する。すなわち、このディメンションは存在するが、制御はされないことを意味する。一つのディメンションは起動することもできれば動作を停止することもできる。ディメンションの動作が停止すると、このディメンションに属する入力されたタスクは始動されないが、このディメンションにおいて実行中であるタスクは、実行され続ける。

【0025】プロセスが、「JRS」タスクの実行コマンドにより生じる場合で、一定のディメンションが「JRS」コマンド内で特定された場合、このプロセスは特定されたディメンションに属する。コマンド名およびユーザが一つのディメンションに関連する先祖（ancestor）をプロセスが有する場合、プロセスはこのディメンションに属する。プロセスの実行時間の間、相対的な重み付けは、ディメンションを含むプロセスの優先順位を管理かつ変更するために用いられる。一つのディメンションにおける全てのプロセスの優先順位も同様の方法で変化する。「ダイバース」ディメンション（MISC）のプロセスは、最大値で、すなわち優先順位が最も低いシステムの負荷が大きい場合、「NICE」を有する。つまり、優先順位は、コンフィギュレーションファイルにおいて管理者が決定する閾値よりも高い。「ユニックス」システムにおいて、「NICE」は、値が優先順位に逆比例するプロセスに関するインジケータである。本発明の制御機構はオプションであって、ディメンションにユーザアプリケーションを記録するには、二つの方法がある。ディメンションの識別子は、タスクのプログラムを介するアプリケーションタスクの入力によって明白に供給することができる。一つのアプリケーションのプロセスは、プロセスとユーザとにより実行されるコマンド名を特定することによって一つのディメンションに関連させられる。プロセスはユーザの名前の下で実行される。本発明による優先順位の動的制御機構は、二個の値を考慮することからなる。最初の値「Pdc」は、ディメンションに属する全てのプロセスによって用いられる資源の現在の割合を示す。第二の値「Pde」は、ディメンションによって用いられるはずの資源の割合を示す。第一の値「Pdc」は、システムのプロセステーブルを定期的に走査し、それらのディメンションにプロセスを関連付け、またディメンションの各プロセスが用いる資源の各ディメンションに対する和を計算することによって得られる。

【0026】第二の値「Pde」は、次の式によって計算される。

【0027】

【数1】

$$Pde = \frac{Rd}{\sum R_i} * Pns$$

【0028】ここで「Rd」は、そのディメンションの相対的な重み付けである。「R_i」は、各ディメンションの相対的な重み付けである。和「 $\sum R_i$ 」は、考慮された時間周期間に作動した全てのディメンションの全ての相対的な重み付けの和である。「Pns」は、「システム」ディメンションに属さず、相対的な重み付けを有し、考慮された時間間隔の間資源が消費されたプロセスが用いる資源の割合である。「Pdc」と「Pde」との差が所定の閾値よりも大きい場合、そのディメンションに属するプロセスの値「NICE」は、いずれかの方向に変えられる。差が大きければ大きいほど、「NICE」値も変えられる。「NICE」値の可変範囲は、0～40である。「NICE」値は、「システム」ディメンションに対して26という値に決められている。「NICE」値が増加すると、優先順位は減少する。たとえばディメンション「MYDIM」を有するシステムにおいて、このディメンションは、ディメンション「ロット」（バッチ）に対して相対重み付け1が宣言されたので、システムで相対重み付け2を持つように決定することができる。システムの負荷が非常に大きい場合、ディメンション「MYDIM」に属するプロセスの優先順位は、次のように変えられる。

【0029】ディメンション「MYDIM」および「ロット」のプロセスが、プロセッサ装置（CPU）の90%を用いている場合、ディメンション「MYDIM」のプロセスの優先順位は、いずれかの方向に変えられ、その結果、ディメンション「MYDIM」によって用いられるCPUの割合は60%に近くなる。

【0030】さらに、「LRM」アプリケーションプログラムインターフェース34は、「LRM_gui」グラフィカルユーザインターフェース32と関連付けされる。このインターフェース32により、ユーザは、マウスまたはマウス以外の他の要素の作動によって、事前にプログラムされたグラフィカルインターフェースにより処理される外部事象をトリガることができる。グラフィカルインターフェースは、たとえばボタンのオンオフ、ドラッグといったこれらの事象およびウィンドウ内へのマウスポインタの入力などの論理的事象を認識することができる。また、このインターフェース32によって、ユーザは、同じくマウスまたはマウス以外の他の要素の作動によってグラフィカルインターフェース内で処理を開始することができ、これによって、図2Aに示された通常のメニューから、セッション、デーモン、資

源、ディメンション、ヘルプからなるメニューの様々な項目を含むウインドウ321をディスプレイさせることができる。このメニューから、マウスによって一つの項目を選択できる。マウスボタンをクリックした後で、グラフィカルインターフェースプログラムによる処理後、項目の選択により、「デーモン」という項目に対して図2Bのようなサブメニューが現れる。このサブメニュー3211は、デーモンの状態の表示、デーモンの始動、デーモンの停止、始動スクリプトのディスプレイ、終了スクリプトのディスプレイ、コンフィギュレーションパラメータのディスプレイを可能にする機能を含むこのサブメニューの項目の一つを、マウスによって強調、あるいは太字化を可能にする。

【0031】マウスによる「資源」項目の選択により、グラフィカルインターフェースプログラムによる処理後、図2Cに示されたようなサブメニュー3212の表示が現れる。このサブメニューは、表示、履歴、閾値といった機能の一つの選択可能性を含む。「表示」機能を選択すると、図2Caに示したように、「jor」ファイルの情報抽出による「LRM gui」インターフェースは、各々がCPUの使用率および、実メモリ、仮想メモリ、一時的なファイルスペースの使用率をそれぞれ示す複数のヒストグラムバーを含むウインドウをディスプレイすることができる。「履歴」機能を選択すると、図2Cbに示したように、「jor」ファイルの情報抽出により「LRM gui」インターフェースは、システムの各項目すなわちCPU、実メモリ、仮想メモリ、一時的なファイルスペースに対して、10秒間周期でシステムの資源消費の最近の履歴を示すことができる。

【0032】図2Dは、メインメニューのバーにおいて「ディメンション」項目を選択した後で行われる表示を示すものである。この選択により、可能な複数機能を含むサブメニューが表示される。最初の機能3213aは追加を可能にし、二番目3213bは表示を、三番目3213cは変更を、四番目3213dはディメンションの削除を、五番目3213eは、ディメンションにより消費される資源を知ることを、六番目3213fは、プロセスを知ることを、七番目3213gは全体的な管理を知ることを可能にする。たとえばユーザーがマウスを用いて第三機能を選択し、マウスボタンをクリックすることによりこれを強調し、有効にすると、「変更」機能が強調される。こうした動作は、グラフィックインターフェースプログラムによる処理後、システムにおいて利用可能な複数のディメンションをウインドウ32131内に含む二番目のサブメニューを表示する。この第二のサブメニューのウインドウにおいて利用可能なディメンションの一つをマウスを用いて選択すると、グラフィカルインターフェースプログラムによる処理後、図2Eの表示が現れる。この表示は、ディメンションの名称を定義可能な第一の文字数字枠3241と、相対的な重み付け

を定義可能な第二の文字数字枠3242と、動的制御機能を起動可能な第三の文字数字枠3242を含む。またウインドウは、別々の二つのコラムで表示可能なスクロールリスト3244を含む。一方のコラムはコマンドに割り当てられ、もう一方はユーザーに割り当てられ、各コマンドおよび各ユーザーは各コマンドに関連付けされている。ユーザーは、マウスを用いて「コマンド」コラムの一個の枠3245または「ユーザー」コラムの一個の枠3246に位置決めし、キーによって実行すべきコマンドを入力するのと同様に、関連するディメンションを「ユーザー」コラムに入力することができる。「確認」ボタン3247および「中止」ボタン3248は、既存のディメンションを変更する情報を確認し、あるいは変更を実施せずに中止することができる。「確認」ボタンを押すと、情報は「jor」ファイルに記憶され、このように入力あるいは変更された相対的な重み付けが、動的制御方法において用いられる。

【0033】図2Fは、メニュー3213で「資源」機能を選択した場合の、ディメンション別資源の使用を示すウインドウの表示を示す。この使用はヒストグラム形式で示され、各ヒストグラムは、各ディメンションに対してシステムの項目の一つすなわちCPU、実メモリ、仮想メモリに関連する。ヒストグラムは、グラフィカルインターフェースプログラムによる処理によって図示される。同様に「プロセス」機能を選択すると、グラフィックインターフェースプログラムによる処理後、「NICE」を用いた最後の十秒間にCPU時間を最も消費したプロセスのリストをウインドウ2Gに表示することができる。また最後の機能「総合管理」を選択すると、図2Hに示したウインドウ内に各ディメンションに対する相対的な重み付けの管理を表示することができる。これによってユーザーは、各ディメンションに関連する各枠内に相対的な重み付けの新しい値を入力し、現在の割合を示す枠内に、現在の割合の変更をディスプレイすることができる。さらに、「変更」機能に対して、「確認」ボタン3263および「中止」ボタン3264により、相対的な重み付けの変更を確認し、あるいは変更を実施せずに機能を中止することができる。

【画面の簡単な説明】

【図1】情報処理システムおよび、本発明の方法を実施可能にするための情報処理システムに関連付けされるソフトウェア手段を示す概略図である。

【図2A】本発明によるローカル資源管理プログラムの開始の際に、スクリーンにディスプレイされるウインドウのメニューバーを示す図である。

【図2B】先行するウインドウのメニューの構成要素の一つを選択した後の、メインメニューを示す図である。

【図2C】メニューの別の構成要素を選択した後の、メインメニューを示す図である。

【図2D】「資源」を構成する要素から「ディスプレ

イ」機能を選択した場合に行われる表示を示す図である。

【図2Cb】「資源」を構成する要素から「履歴」機能を選択した後のスクリーン表示を示す図である。

【図2D】「変更」機能を「ディメンション」構成要素から選択した後の、サブメニューを示す図である。

【図2E】図2Dでの選択後に実施される表示を示す図である。

【図2F】「ディメンション」メニューから「資源」機能を選択した後で実施される表示を示す図である。

【図2G】「ディメンション」メニューから「プロセス」機能を選択した後に実施される表示を示す図である。

【図2H】「ディメンション」構成要素から「総合管

理」機能を選択した後に実施される表示を示す図である。

【符号の説明】

1 資源

2 「ユニックス」カーネル

3 LRM (ローカル資源管理) デーモン

5 jor ファイル

6 カタログ

31 LRM共有メモリ

32 LRM gu i

33 LRMコンフィギュレーションファイル

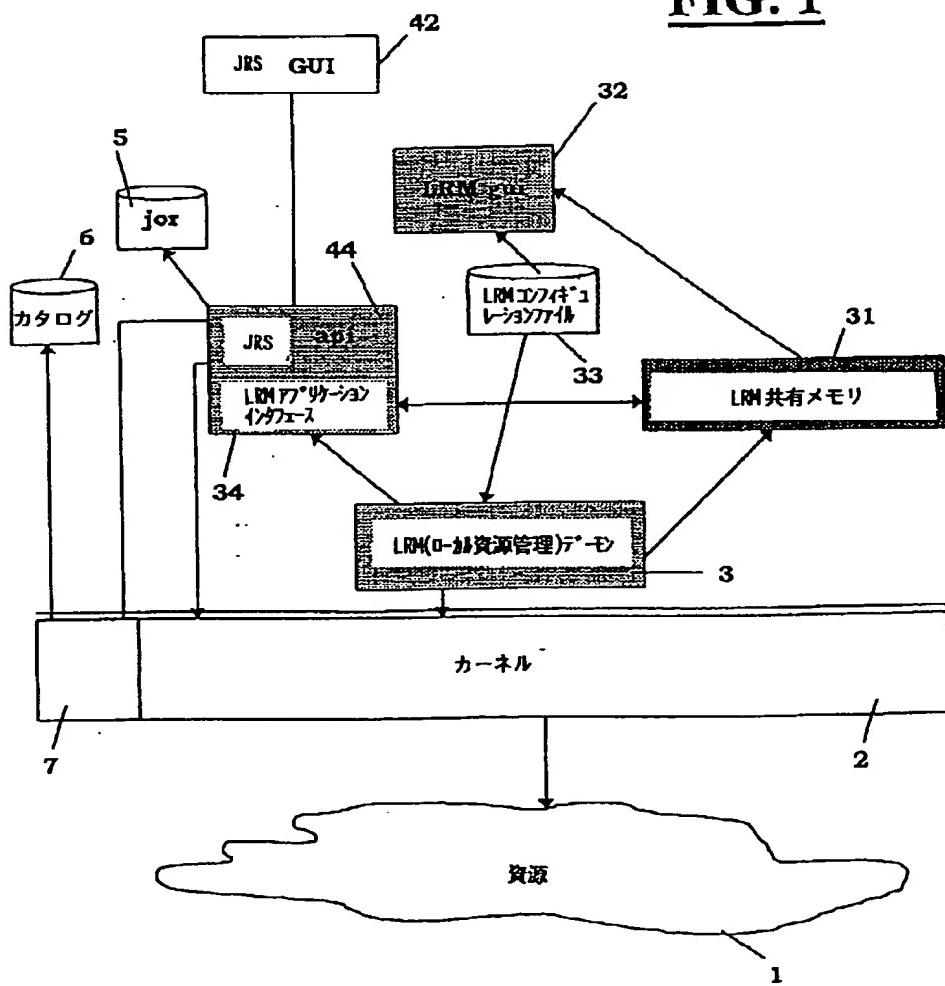
34 LRMアプリケーションインターフェース

42 JRS GUI

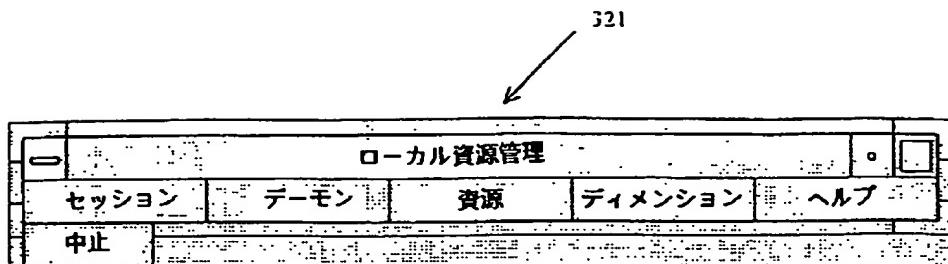
44 JRS アプリケーションインターフェース

【図1】

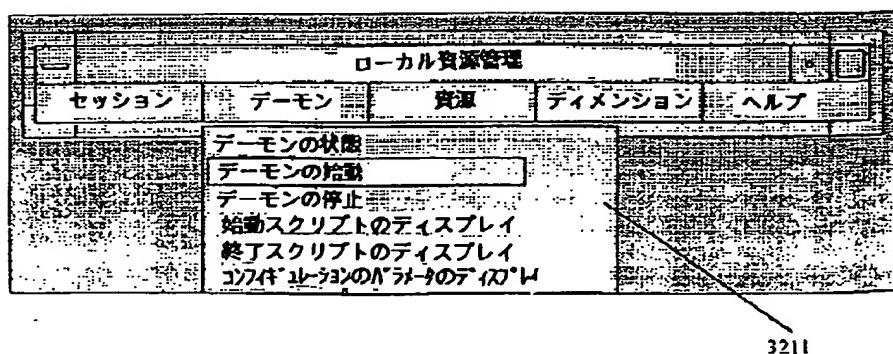
FIG. 1



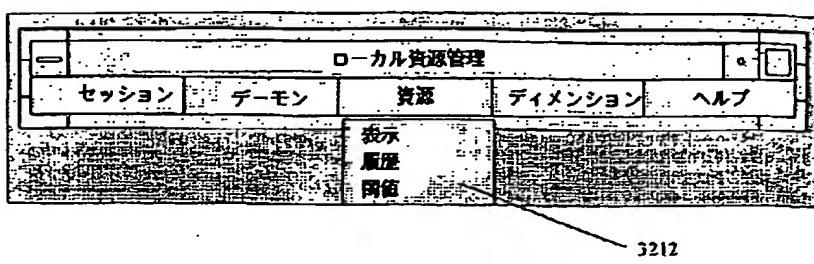
【図2A】

FIG. 2A

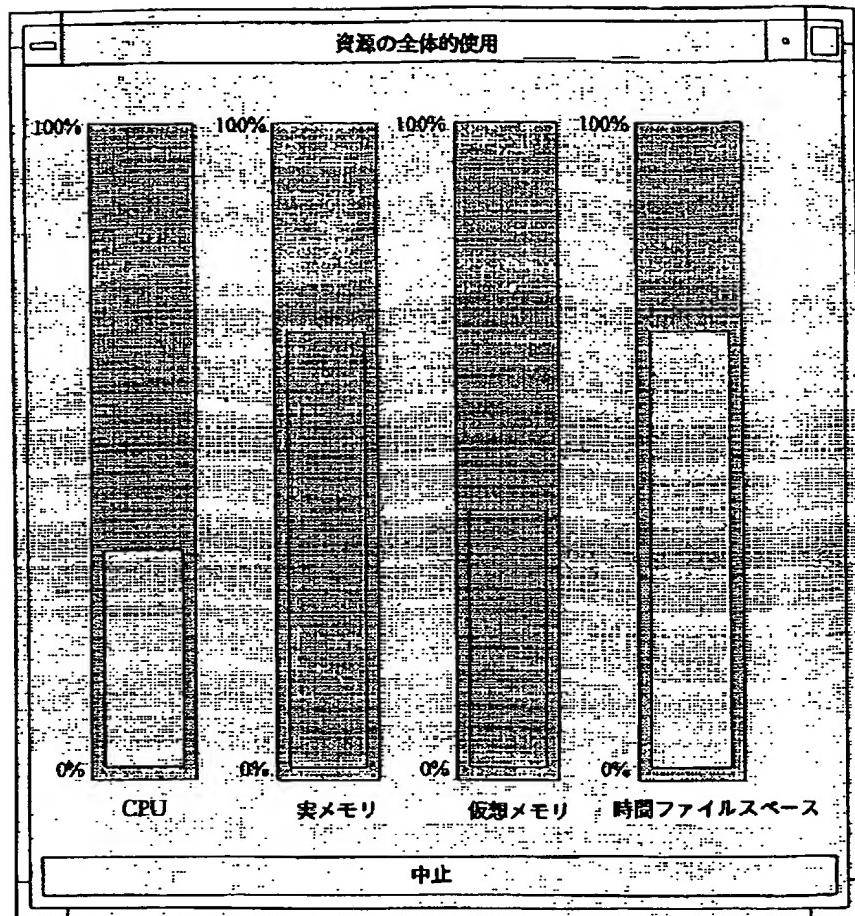
【図2B】

FIG. 2B

【図2C】

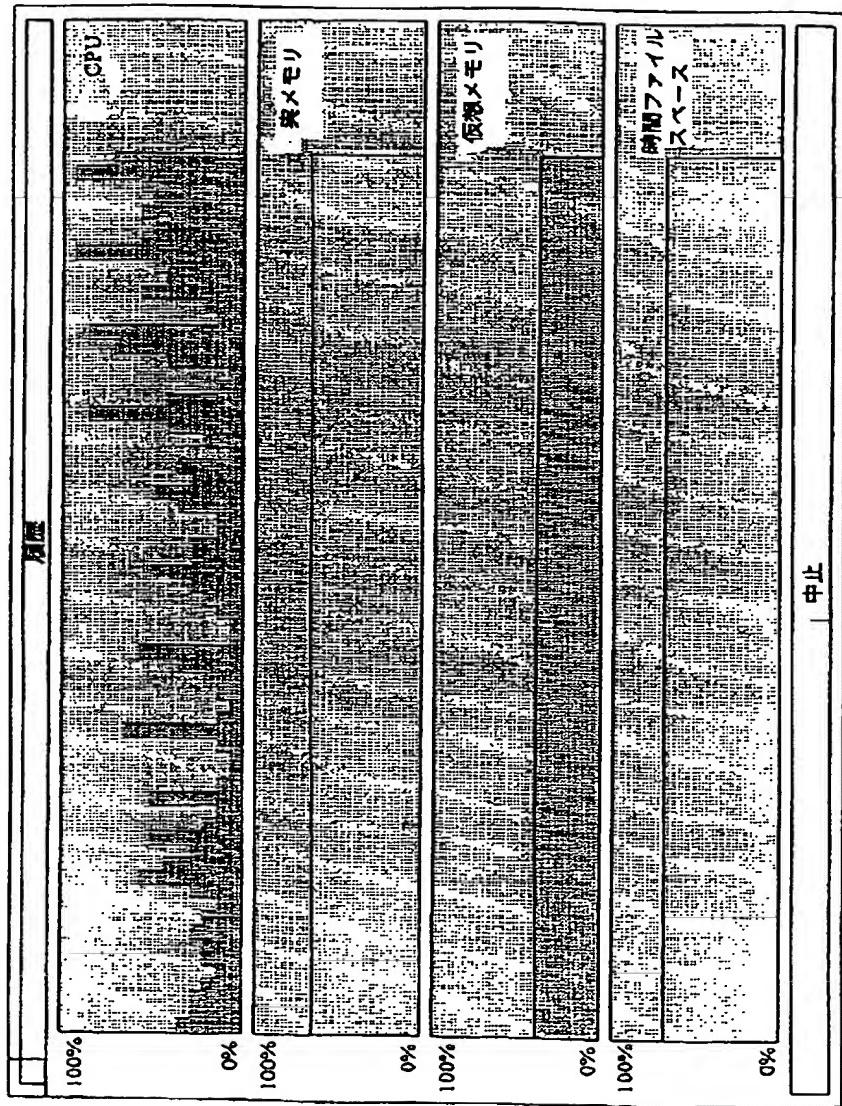
FIG. 2C

【図2Ca】

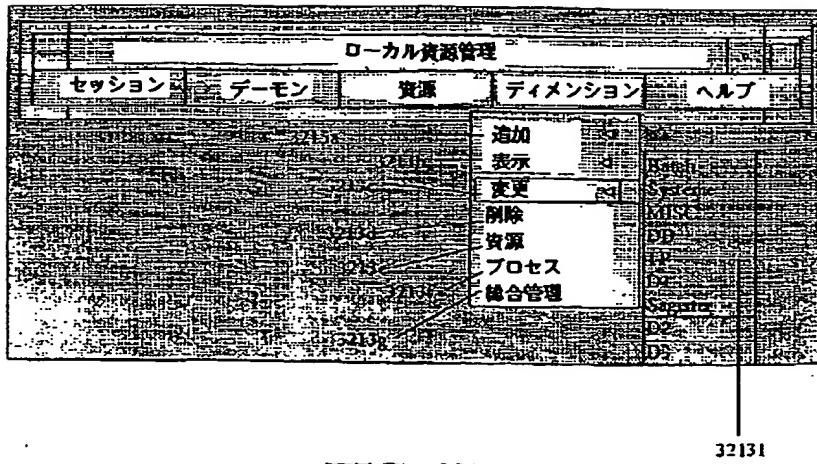
FIG. 2Ca

【図2Cb】

FIG. 2Cb



【図2D】



32131

FIG. 2D

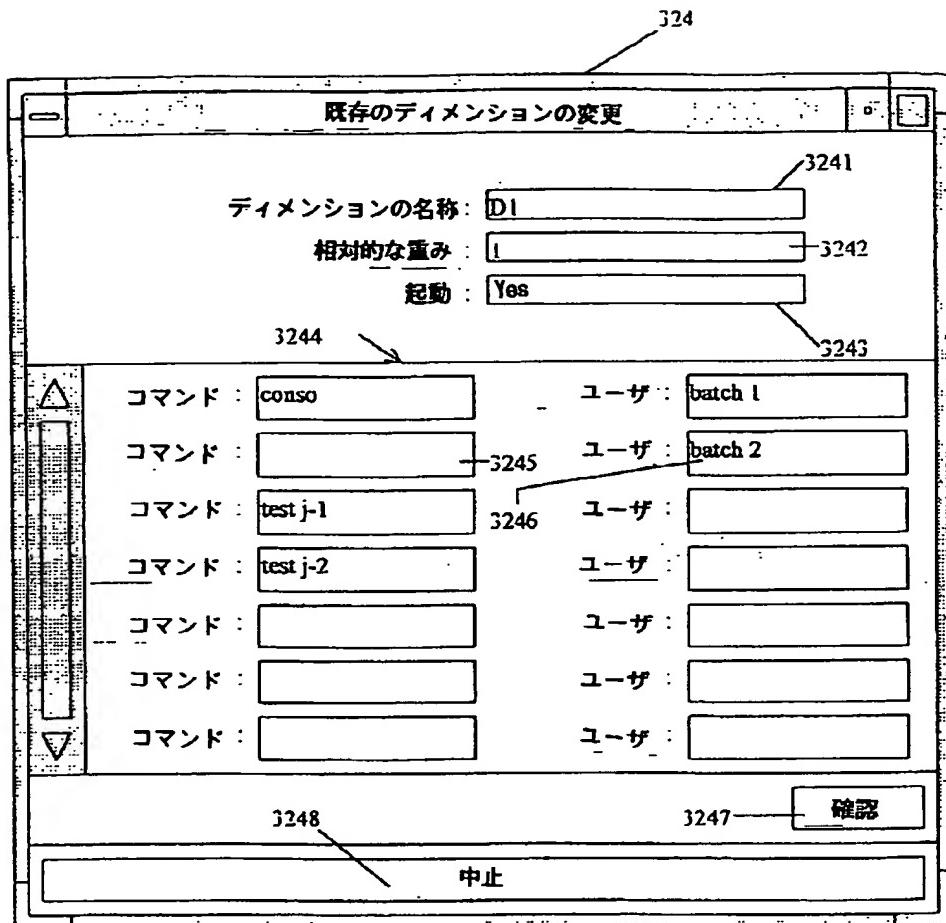
【図2G】

現在動作中のプロセスによる資源利用												
Utl	Pid	コマンド	ティメンション	タスク	開始時刻	Uti Cpu	Sys Cpu	CPU%	実メモリ	仮想メモリ	差	
geo	15722	X	Système	-	09:58:24	1.26s	0.70s	19.60	1365	1077	20	
geo	15272	maker 3X.exe	Framemaker	-	09:58:41	0.48s	0.57s	10.50	1756	1063	20	
root	1	Init	Système	-	14:40:42	0.97s	0.03s	10.00	65	76	20	
root	29936	LRMDémon	LRMD	-	14:49:31	0.12s	0.29s	4.10	419	357	0	
root	46962	Visit	LRMD	-	15:25:42	0.07s	0.06s	1.30	403	205	0	
geo	40344	mwm	Système	-	09:58:38	0.03s	0.04s	0.70	266	172	20	
root	1032	gil	Système	-	14:34:05	0.00s	0.05s	0.50	10	10	41	
root	24150	StatUtil	StatUtil	-	14:45:02	0.00s	0.05s	0.50	164	191	0	
root	50904	LRMinfo	LRMD	-	15:51:12	0.00s	0.05s	0.50	30	15	0	
geo	10240	xterm	Système	-	14:46:24	0.02s	0.01s	0.30	202	113	20	
root	3480	asterix	Système	-	17:00:04	0.01s	0.00s	0.10	183	93	20	
geo	14426	xv	Système	-	15:24:54	0.01s	0.00s	0.10	1184	1029	20	
geo	19070	xdock	Système	-	09:58:36	0.00s	0.01s	0.10	96	58	20	
root	23368	mogind	Système	-	14:46:25	0.00s	0.01s	0.10	95	72	20	
geo	46412	mogin	Système	-	14:43:25	0.00s	0.01s	0.10	9516	38	20	

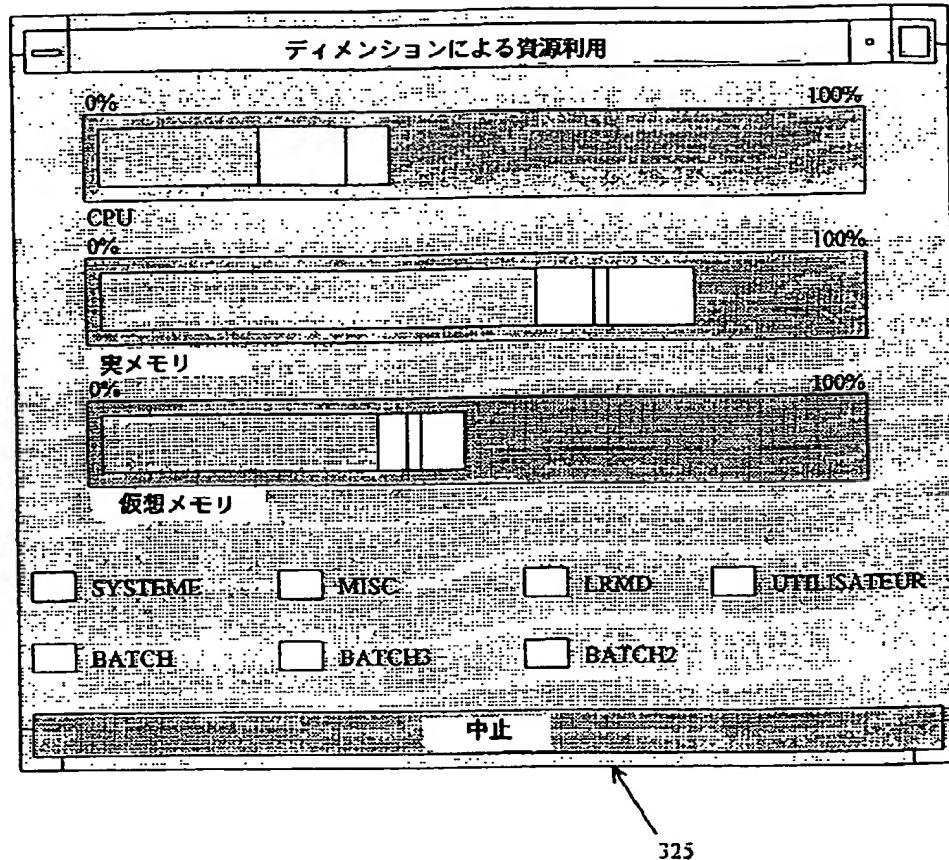
中止

FIG. 2G

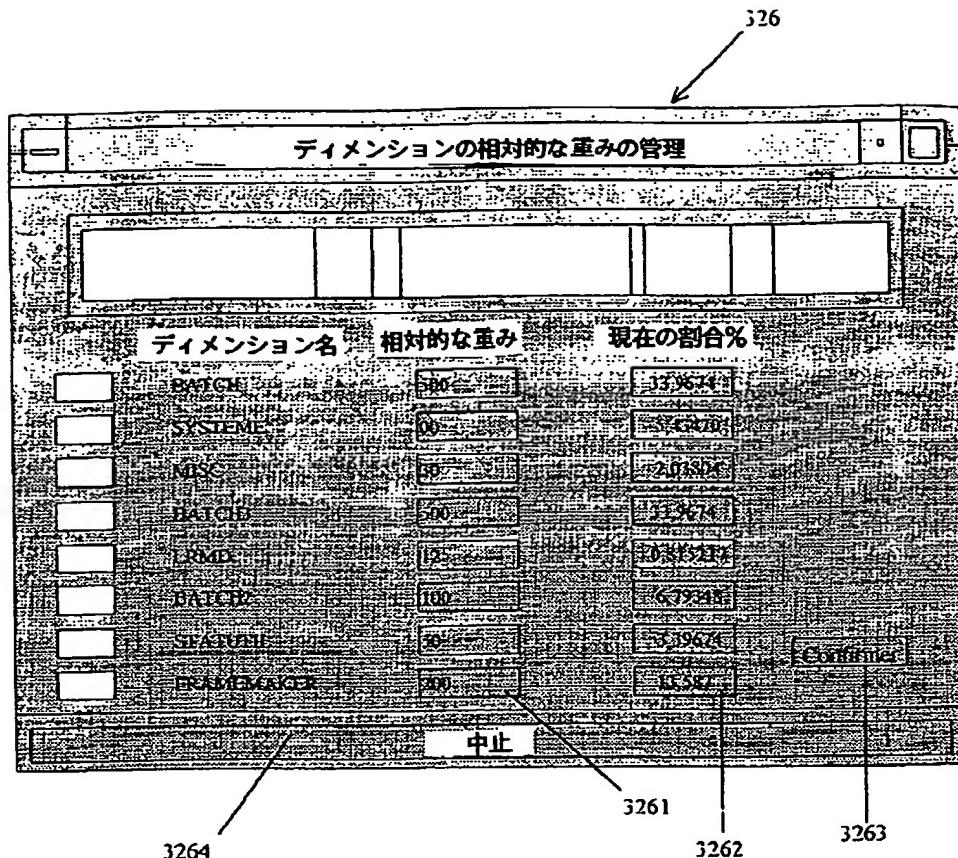
【図2E】

FIG. 2E

【図2F】

FIG2F

【図2H】

**FIG. 2H**